

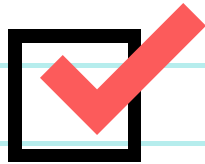
DESARROLLO WEB
PHP

ACTIVIDAD 3

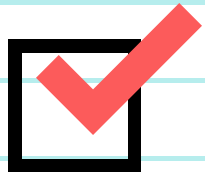
PRESENTADO POR: ENZO
REYES H.



CONTENIDO

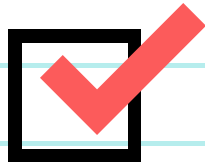


**RAZONES PARA
USAR FUNCIONES**



**COMUNIDADES
VIRTUALES DE AYUDA**

DESARROLLO



**RAZONES PARA
USAR FUNCIONES**



**COMUNIDADES
VIRTUALES DE AYUDA**

```
if ($hayCafe==true){$tomar  
}$else{$hacerCafe}
```



RAZÓN

01

SIMPLIFICAR EL CÓDIGO

ES DECIR:

Usar
funciones
nos evita el
poner
muchas
veces la
misma
instrucción
iterativa.

```
="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
ef="/favicon.ico" type="image/x-icon">
con.ico" type="image/x-icon">

stylesheet" href="css/materialize.min.css" media="screen, projection"
f="https://maxcdn.bootstrapcdn.com/font-awesome/4.6.0/css/font-awesome.min.css"
ef="/css/animate.css">
ef="css/theme.css">

apper">
tainer">
class="brand-logo hide-on-med-and-up"><h1><span style="font-weight: bold;">
w">
m8 18 hide-on-small-only">
ight:400">starter</span><span style="font-weight: bold;">
tup network and workspace trail for team
span id="banner-creator" class="waves-effect waves-light" style="border: 1px solid #ccc; padding: 5px; display: inline-block; text-align: center; width: 100px; height: 30px; line-height: 30px; vertical-align: middle;">

```

“

Ejemplo de código que se repite

```
<?php
//Notas Estudiante 1
$nota1=4;
$nota2=3.5;
$nota3=4.5;
$nota4=4.8;
$promedioEstudiante1=($nota1+$nota2+$nota3+$nota4)/4;
//Notas Estudiante 2
$nota1=2.8;
$nota2=3.1;
$nota3=3;
$nota4=2.5;
$promedioEstudiante2=($nota1+$nota2+$nota3+$nota4)/4;
//Notas Estudiante 3
$nota1=5;
$nota2=5;
$nota3=4.5;
$nota4=4.8;
$promedioEstudiante3=($nota1+$nota2+$nota3+$nota4)/4;
?>
```

Ejemplo de código en una función

```
$promedioEstudiante1=promedio(4,3.5,4.5,4.8);

function promedio($nota1,$nota2,$nota3,$nota4) {
    return ($nota1+$nota2+$nota3+$nota4)/4;
}
```

En ambos casos se cumple el objetivo, pero en el segundo caso, se evita repetir código.

EJEMPLO PRÁCTICO

RAZÓN 02

MODULARIDAD

Una función permite a nuestro código ser particionado para obtener límites bien definidos y por sobre todo mejora la comprensión del mismo.

Bertrand Meyer



“

```
$promedioEstudiante1=promedio(4,3.5,4.5,4.8);
```

```
function promedio($nota1,$nota2,$nota3,$nota4) {  
    return ($nota1+$nota2+$nota3+$nota4)/4;  
}
```

OTRO EJEMPLO

Si un valor en una función
son números, por obligación
los demás símiles serán
números

ESTANDARIZACIÓN

RAZÓN 03

#FUNCIONES()



REUSO

RAZÓN 04

**Al igual que este slide
una función permite
reutilizarse las veces
que sean necesarias**

RAZÓN 04

“

```
function promedio($nota1,$nota2,$nota3,$nota4) {  
    return ($nota1+$nota2+$nota3+$nota4)/4;  
}
```

```
$promedioEstudiante1=promedio(4,3.5,4.5,4.8);
```

```
$promedioEstudiante2=promedio(2.8,3.1,3,2.5);
```

```
$promedioEstudiante3=promedio(5,5,4.5,4.8);
```

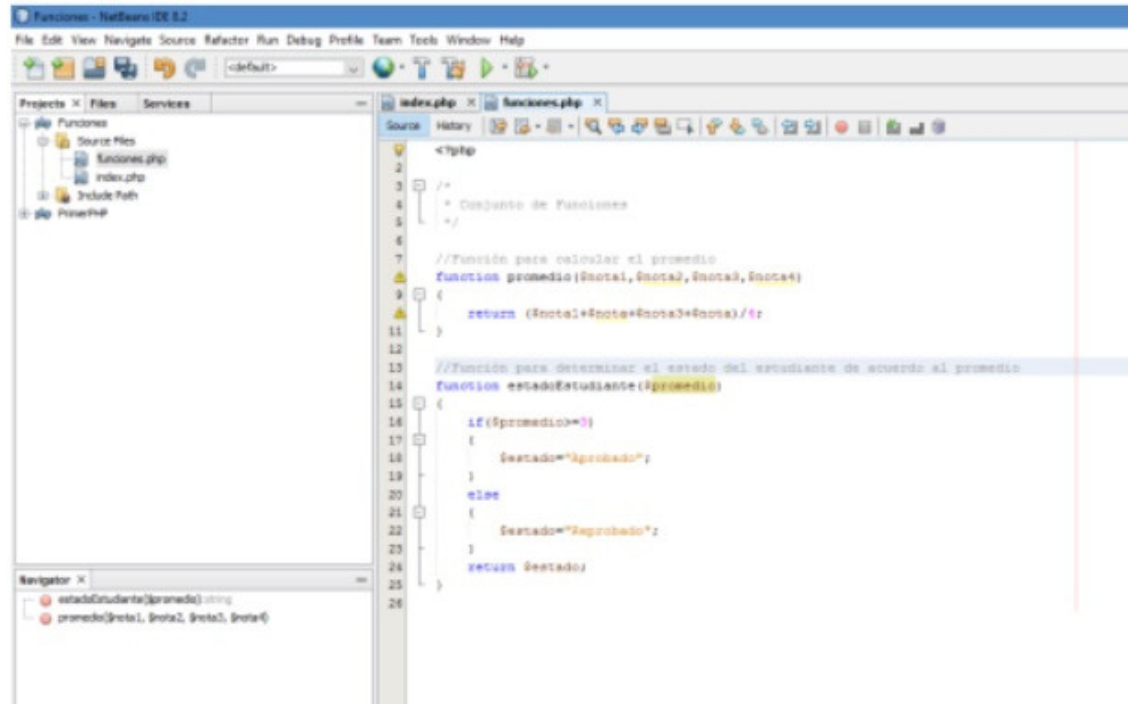
**EJEMPLO: LA MISMA FUNCIÓN
CON DIFERENTES VALORES**

RAZÓN 05

COLECCIONES

TODAS LAS FUNCIONES QUE GENEREMOS LAS PODREMOS
ALMACENAR EN UNA O MUCHAS BIBLIOTECAS PARA QUE
PUEDAN SER INVOCADAS DESDE CUALQUIER PARTE DE
NUESTRO PROYECTO POR MEDIO DEL ELEMENTO
INCLUDE() O REQUIRE ()

“



The screenshot shows the NetBeans IDE interface. The main editor window displays a PHP file named 'funciones.php'. The code defines two functions: 'promedio' and 'estadoEstudiante'. The 'promedio' function calculates the average of four grades. The 'estadoEstudiante' function determines the student's status based on the average. The Navigator panel on the left shows the project structure, including 'Funciones' and 'index.php'.

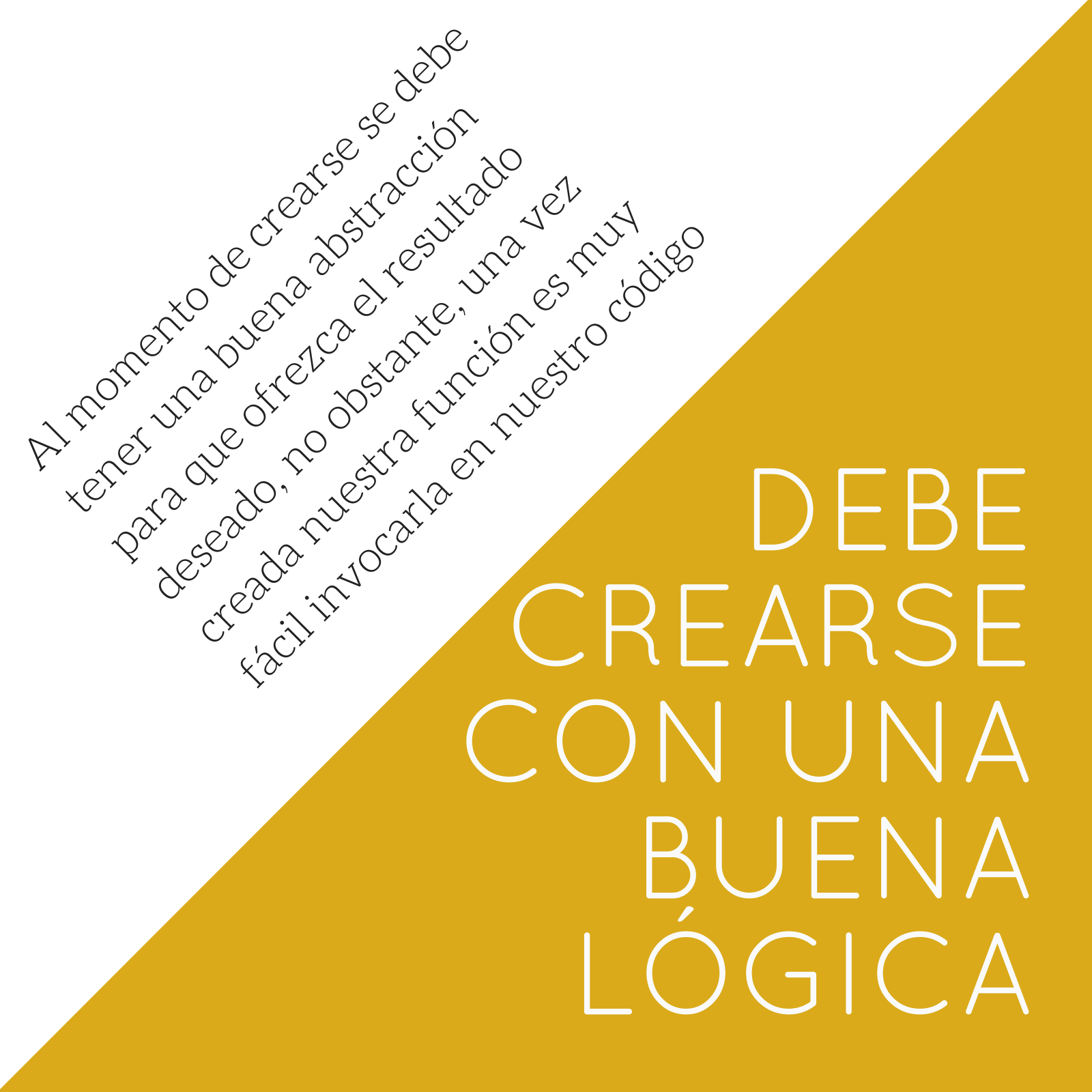
```
<?php
2
3 /*
4  * Conjunto de Funciones
5  */
6
7 //Función para calcular el promedio
8 function promedio($nota1, $nota2, $nota3, $nota4)
9 {
10     return ($nota1+$nota2+$nota3+$nota4)/4;
11 }
12
13 //Función para determinar el estado del estudiante de acuerdo al promedio
14 function estadoEstudiante($promedio)
15 {
16     if ($promedio >= 6)
17     {
18         $estado = "Aprobado";
19     }
20     else
21     {
22         $estado = "Reprobado";
23     }
24     return $estado;
25 }
26
```

**EJEMPLO: COLECCIÓN DE
FUNCIONES**

RAZÓN 06

SON FÁCILES DE EMPLEAR Y
RÁPIDAS DE INVOCAR





Al momento de crearse se debe
tener una buena abstracción
para que ofrezca el resultado
deseado, no obstante, una vez
creada nuestra función es muy
fácil invocarla en nuestro código

DEBE
CREARSE
CON UNA
BUENA
LÓGICA

“

```
function sumar($numero1, $numero2){  
    echo "La suma es: " . ($numero1 + $numero2);  
}  
  
//probando la funcion  
sumar(2, 6);
```

**EJEMPLO DE LÓGICA E
INVOCACIÓN DE UNA
FUNCIÓN**



RAZÓN 07

SON GLOBALES

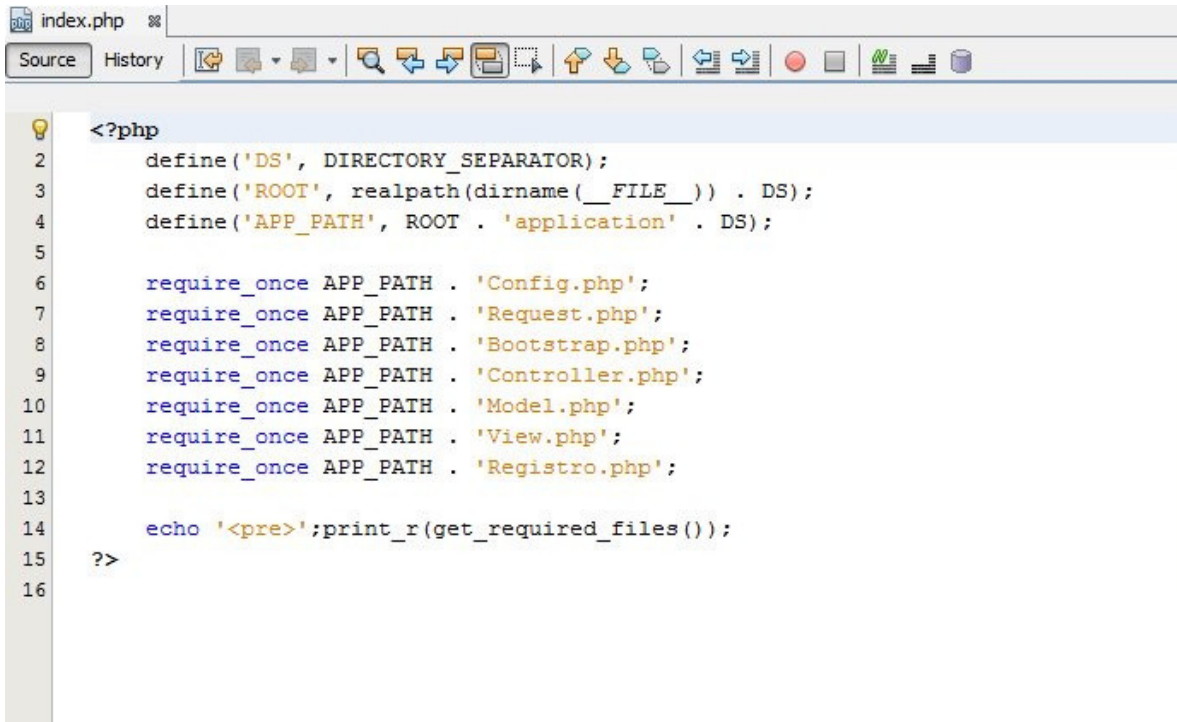


RAZÓN 07

**AL SER ALMACENADAS
EN BIBLIOTECAS PUEDEN
VOLVERSE PLANTILLAS
PARA FUTUROS
PROYECTOS**

ES POR ESO QUE SE PUEDEN VER LOS FAMOSOS
FRAMEWORKS TAL COMO "SYMFONY"

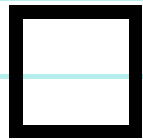
“



```
index.php %
Source History
<?php
2  define('DS', DIRECTORY_SEPARATOR);
3  define('ROOT', realpath(dirname(__FILE__)) . DS);
4  define('APP_PATH', ROOT . 'application' . DS);
5
6  require_once APP_PATH . 'Config.php';
7  require_once APP_PATH . 'Request.php';
8  require_once APP_PATH . 'Bootstrap.php';
9  require_once APP_PATH . 'Controller.php';
10 require_once APP_PATH . 'Model.php';
11 require_once APP_PATH . 'View.php';
12 require_once APP_PATH . 'Registro.php';
13
14 echo '<pre>';print_r(get_required_files());
15 ?>
16
```

**EJEMPLO DE UN FRAMEWORK
BÁSICO**

DESARROLLO



**RAZONES PARA
USAR FUNCIONES**



**COMUNIDADES
VIRTUALES DE
AYUDA**

`require ("TOMARCAFE.PHP");
require ("GALLETAS.PHP");`



UNA COMUNIDAD HECHA POR
DESARROLLADORES PARA DESARROLLADORES



PROPÓSITO

ENTREGAR CONOCIMIENTO AL MUNDO

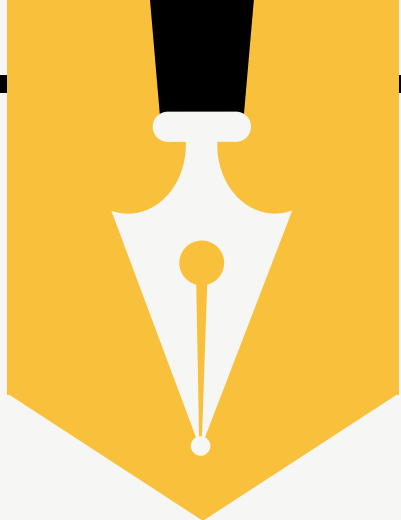
Esta es una comunidad hecha para compartir, departir, debatir, preguntar sobre diversos temas en programación bajo diversos lenguajes, la desventaja de esta comunidad es que la mayoría de su comunidad es de habla inglesa, no es apta para quienes no conocen del idioma.

COMUNIDAD

GitHub



UNA COMUNIDAD HECHA PARA SOLUCIONAR
MUCHOS DE TUS PROBLEMAS



PROPÓSITO

DESARROLLAR PROYECTOS EN CONJUNTO

Esta comunidad a diferencia de la primera goza de una serie de repositorios de los cuales un desarrollador puede echar mano a su vez, quienes desarrollan pueden subir sus ideas o trabajos para que puedan ser potenciados por otros miembros de la comunidad, ejemplo de ellos es Atom, el editor de código hackeable.

la desventaja es que para su uso ya se deben de tener bases de desarrollo para poder comprender y aprovechar sus recursos

COMUNIDAD

phpcentral
com



UNA COMUNIDAD HECHA PARA COMPARTIR
CONOCIMIENTO CON EL MUNDO



PROPÓSITO

COMPONER LA COMUNIDAD HISPANOHABLANTE MAS GRANDE EN DESARROLLO PHP

Ideada como una red social, esta comunidad permite al usuario crear sus entradas con preguntas o trozos de código para que sean revisados y mejorados, su desventaja es que al estar en maduración es probable que tus dudas no sean resueltas a la inmediatez.

¡MUCHAS GRACIAS!

ALGUNAS PARTES DE ESTE MATERIAL FUERON EXTRAÍDAS
DE: [HTTPS://APP.EMAZE.COM/USER/JMDIAZPU](https://app.emaze.com/user/jmdiazpu), WIKIPEDIA.COM, Y
COMUNIDADES CONSULTADAS

EL CONOCIMIENTO ES PARA TODOS, POR LO CUAL:

